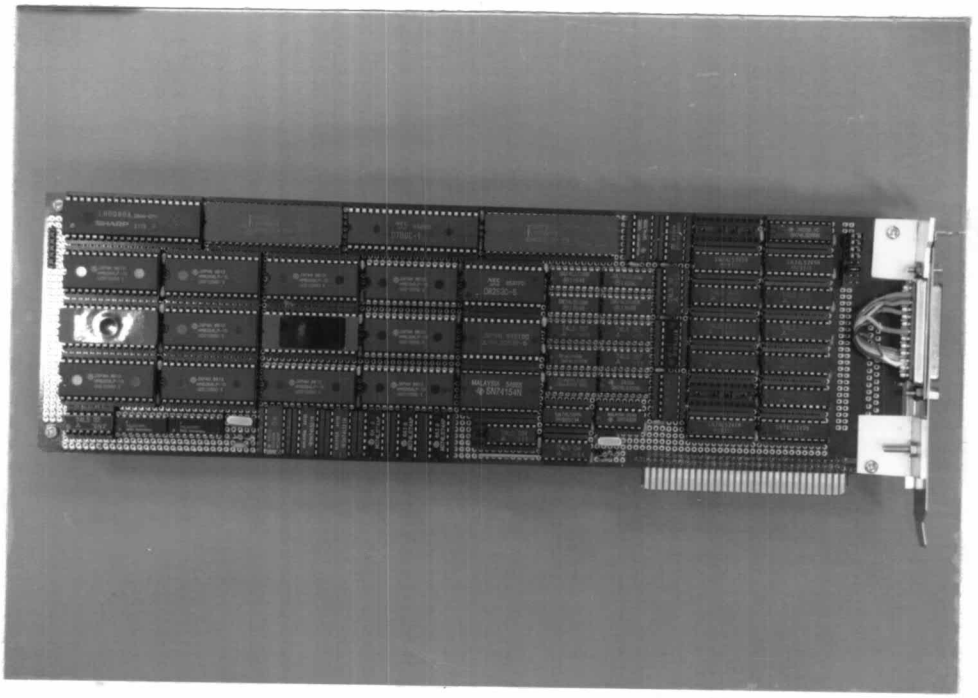
บทที่ 5

การทดสอบและผลการทดสอบ

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

5.1 การทดสอบในส่วนฮาร์ดแวร์

ฮาร์ดแวร์ที่ใช้ในการตรวจสอบโพรโทคอลชนิด SDLC แสดงดังรูปที่ 5.1 ในหัวข้อนี้จะเป็นการทดสอบหน่วยความจำ, พอร์ทต่าง ๆ และบัฟเฟอร์



รูปที่ 5.1 ฮาร์ดแวร์ที่ใช้ในการตรวจสอบโพรโทคอลชนิด SDLC

5.1.1 การทดสอบหน่วยความจำ เนื่องจากหน่วยความจำแบ่งเป็น 2 ส่วนคือ ส่วนที่ต่ออยู่กับบัฟเฟอร์และส่วนที่ไม่ต่ออยู่กับบัฟเฟอร์ ดังนั้นการทดสอบนี้จึงกำหนดให้บัฟเฟอร์แทนบัส-

แอดเดรสและบัสข้อมูลจากซีพียู Z-80 เท่านั้น ในกรณีนี้Z-80 จะอ้างแอดเดรสได้ทั้งหมดคือในช่วง 0000H-BFFFH การทดสอบกระทำโดยเขียนข้อมูลลงไปในหน่วยความจำและอ่านข้อมูลจากหน่วยความจำนั้นขึ้นมาเปรียบเทียบกับข้อมูลเดิม หลังจากนั้นก็นำข้อมูลในหน่วยความจำขึ้นมาแสดงผลบนจอภาพทุก ๆ ไบต์จนครบ 48 กิโลไบต์ของแต่ละด้าน ขั้นต่อมาเป็นการเปลี่ยนการกำหนดสถานะของบัฟเฟอร์ใหม่ในกรณีนี้จะทำให้ Z-80 อ้างแอดเดรสได้เฉพาะ 0000H-1FFFH และ A000H-BFFFH เท่านั้นส่วนการทดสอบก็ทำในลักษณะเดียวกันซึ่งผลที่ได้เป็นไปตามที่ออกแบบไว้

5.1.2 การทดสอบพอร์ทต่างๆ พอร์ทที่ทดสอบในที่นี้คือ 8253 และ 8273 ในที่นี้กำหนดให้ 8253 มีโมดการทำงานที่เรียกว่า hardware trigged strobe ข้อมูลที่เขียนลงไปในรีจิสเตอร์ของ 8253 มีทั้งหมด 3 ไบต์ โดยไบต์แรกเป็นคำสั่งกำหนดการทำงานให้ 8253 ส่วนสองไบต์ที่เหลือใช้เป็นค่าคงที่ในการนับ เมื่อดำเนินโปรแกรมนี้และตรวจสอบการทำงานของ 8253 โดยเทียบกับสัญญาณนาฬิกาที่ใช้ในการนับ 8253 นับค่าได้ถูกต้องตามที่กำหนดให้การทดสอบ 8273 ในที่นี้ทดสอบได้เฉพาะผลตอบที่มีต่อการเขียนคำสั่งกำหนดฟังก์ชันการทำงานให้ 8273 โดยตรวจสอบได้จาก status register ซึ่งอยู่ภายใน 8273 เอง ผลคือ 8273 สามารถรับคำสั่งและพารามิเตอร์ได้ทุกค่าตามที่กำหนดให้

5.1.3 การทดสอบบัฟเฟอร์และสัญญาณควบคุมบัฟเฟอร์ สัญญาณที่ใช้ในการควบคุมบัฟเฟอร์ได้มาจากการถอดรหัสผ่านไอซีเบอร์ 74LS154 โดยจะไปควบคุมขีดความสามารถในการอ้างแอดเดรสระหว่างซีพียู Z-80 กับ 8088 หลังจากนั้นได้ทดสอบบัฟเฟอร์โดยการย้ายข้อมูลจากหน่วยความจำของซีพียู Z-80 ไปเก็บในหน่วยความจำของ 8088 แล้วนำขึ้นแสดงบนจอภาพจนครบ 32 กิโลไบต์ทั้งสองด้านซึ่งผลการทดลองเป็นไปตามที่ออกแบบไว้

5.2 การทดสอบในส่วนซอฟต์แวร์

ซอฟต์แวร์ที่ใช้ในการทดสอบนี้แบ่งเป็น 2 ส่วนคือ ส่วนที่เป็นภาษาแอสเซมบลีกับส่วนที่เป็นภาษาเทอร์โบปาสกาล

5.2.1 ภาษาแอสเซมบลี ซอฟต์แวร์ในส่วนนี้ใช้ในการแยกเฟรมของ SDLC ออกเป็นเขตต่าง ๆ แล้วจึงนำไปเก็บในหน่วยความจำ นอกจากนี้ยังใช้ในการตรวจสอบสถานะระหว่างซีพียูทั้งสองตัว เคลื่อนย้ายหน่วยความจำ การทดสอบกระทำโดยให้ซีพียู Z-80 ดำเนินโปรแกรมในส่วนนี้ในขณะที่ 8273 ทำการรับข้อมูลอยู่ ผลการทดสอบในขั้นแรกมีข้อผิดพลาดของโปรแกรมซึ่งเกิดจากการเก็บค่าลงในสแตก (stack) ไม่ครบ, การกระโดดไปในตำแหน่งที่ไม่เหมาะสม, การเรียกใช้รีจิสเตอร์ซ้ำซึ่งมีผลทำให้ค่าในรีจิสเตอร์เปลี่ยนแปลง หลังจากมีการแก้ไขโปรแกรมในส่วนนี้หลายครั้งจึงได้โปรแกรมที่ทำงานได้ตามที่ต้องการ

5.2.2 ภาษาเทอร์โบปาสกาล ซอฟต์แวร์ในส่วนนี้นอกจากจะใช้สำหรับแสดงผลแล้วยังใช้ในการควบคุมการทำงานของระบบด้วยโปรแกรมได้ถูกเขียนเป็นกระบวนความย่อย ๆ เพื่อให้สะดวกต่อการแก้ไขหลังจากนั้นจึงนำมารวมเป็นโปรแกรมหลักที่ใช้ควบคุมทั้งระบบ ส่วนที่มีปัญหาของโปรแกรมนี้คือ กระบวนความที่ใช้ในการตอบรับสัญญาณขัดจังหวะที่มาจากซีพียู Z-80 โดยมีสาเหตุมาจากการเก็บค่าลงสแตกไม่ครบนั่นเอง ในส่วนอื่น ๆ ก็สามารถดำเนินได้ตามปกติ

5.3 ผลการทดสอบระบบ

หลังจากทดสอบในส่วนของฮาร์ดแวร์และซอฟต์แวร์ซึ่งสามารถทำงานตามต้องการได้แล้วจึงนำทั้งสองส่วนนี้มารวมกันเพื่อทดสอบระบบทั้งหมด การทดสอบกระทำโดยส่งรูปแบบของโพรโทคอล SDLC ซึ่งมีข้อมูลคือตัวอักษร "A" จำนวน 1,024 ตัวและทำการส่งอย่างต่อเนื่อง ปัญหาที่เกิดขึ้นคือ ซีพียู2 อ่านสถานะของซีพียู1ผิด จากการวิเคราะห์พบว่าช่วงเวลาในการอ่านสถานะของซีพียู2 มากเกินไปทำให้อ่านข้อมูลผิดเข้าไปประมวลผล ดังนั้นจึงเปลี่ยนให้ซีพียู2 มีช่วงเวลาในการอ่านข้อมูลสั้นลงและทำการทดลองใหม่ผลปรากฏว่าเครื่องมือตรวจสอบโพรโทคอลสามารถตรวจสอบและแสดงผลได้ถูกต้องไม่ว่าจะเป็นการแสดงผลในแบบใด (Layer1 หรือ Layer2) ต่อจากนั้นได้ทำการส่งข้อมูลที่ไม่ครบเฟรมทำให้เกิดสถานะ abort หรือทำให้มีสภาวะ idle เกิดขึ้น เครื่องมือตรวจสอบนี้ก็สามารถแสดงผลได้อย่างถูกต้อง การทดสอบต่อมาเป็นการทดสอบกับเครื่องวิเคราะห์โพรโทคอล โดยกำหนดให้เครื่องวิเคราะห์โพรโทคอลจำลองตัวเองเป็น DTE ที่สามารถส่งโพรโทคอลชนิด SDLC ได้ ขั้นตอนที่ใช้ในการทดสอบคือ กำหนดให้ DTE

ส่งคำสั่ง SNRM 5 เฟรมไปยัง DTE อีกข้างหนึ่ง (ในที่นี้มี DTE เพียงข้างเดียว) ถ้า DTE ไม่ได้รับเฟรมตอบกลับจาก DTE อีกข้างหนึ่งก็ให้ส่งคำสั่ง DISC ไปอีก 5 เฟรมและเลิกการติดต่อทันที จากนั้นจึงนำเครื่องมือตรวจสอบโพรโทคอลที่สร้างขึ้นไปทำการตรวจสอบขั้นตอนดังกล่าว ซึ่งผลที่ได้แสดงให้เห็นว่าเครื่องมือตรวจสอบโพรโทคอลที่สร้างขึ้นมานี้สามารถตรวจสอบโพรโทคอลชนิด SDLC ได้อย่างถูกต้อง ในส่วนของการบันทึกและพิมพ์ผลข้อมูลก็เช่นเดียวกัน ดังนั้นจะเห็นได้ว่าเครื่องมือตรวจสอบโพรโทคอลสามารถ ทำงานได้ตามที่ออกแบบไว้ทุกประการ